Free Code Camp Angular 5

By Jeff Maxwell

TCN Solutions Inc.

jeff@tcnsolutions.com

www.jeffmaxwell.com

Other Videos

- https://www.twitch.tv/techlahoma
- Intro to Angular – Maegan Womble – Feb. 4th, 2018
- Intro to Angular Routing – Patrick Hartley – Feb. 11th 2018

Agenda

- Changes
- Classes
- Components
- Modules
- Services
- Performance

Changes

Brief History of Angular(JS)

- AngularJS was released in 2009/2010
- Current Version of AngularJS 1.6 released in Feb. 2018

- Angular 2 - announced in April 2015 and released in Sept. 2016
- Angular 3 – No where to be found???
- Angular 4 – released March 2017
- Angular 5 – released Nov. 2017

Changes to the GOT Routing Project

- Upgraded Angular from 4.2 to 5.2
- Add all the code Patrick did on Feb. 11th

Classes

Typescript Class

```typescript
export class Person {
    _id: string;
    male: boolean;
    house: string;
    imageLink: string;
    slug: string;
    name: string;
    __v: number;
    pageRank: number;
    books: string[];
    updatedAt: Date;
    createdAt: Date;
    titles: string[];
}
```

GOT API - https://api.got.show/api/characters/byId/56ffc5bf0432440819385b09


```
{
  “message”: "Success",
  "data": {
    "_id": "56ffc5bf0432440819385b09",
    "dateOfBirth": 283,
    "imageLink": "/misc/images/characters/Jon_Snow.jpeg",
    "male": true,
    "culture": "Northmen",
    "house": "House Stark",
    "slug": "Jon_Snow",
    "name": "Jon Snow",
    "__v": 0,
    "pageRank": 300,
    "hasPath": true,
    "books": [
      "A Game of Thrones",
      "A Clash of Kings",
      "A Storm of Swords",
      "A Feast for Crows",
      "A Dance with Dragons”
    ],
    "updatedAt": "2016-04-02T13:14:39.685Z",
    "createdAt": "2016-04-02T13:14:39.685Z",
    "titles": [
      "Lord Commander of the Night&apos;s Watch”
    ]
  }
}
```

Components

Add Component

$ ng generate component people-card

OR

$ ng g c people-card

Creates 4 files:

- people-card.component.css – Stylesheet for the page
- people-card.component.html – HTML layout
- people-card.component.spec.ts – Spec Tests
- people-card.component.ts - Typescript Code

Adds entry in @NgModule declarations for PeopleCardComponent

Pass Parameters to Child Components

```
<div *ngFor="let character of characters">
    <app-people-card [person]="character"></app-people-card>
</div>
```

```
export class PeopleCardComponent implements OnInit {
@Input() person: Person;

  constructor() { }

  ngOnInit() { }
}
```

Services

API

- https://api.got.show

Http versus HttpClient (v4.3+)

- JSON as Default
- Interceptors
- Progress Events

Http versus HttpClient (v4.3+)

Http

```
import { HttpModule } from '@angular/http';

Import { Http } from '@angular/http';

return this.http.get(url).map(res => res.json());
```

HttpClient

```
import { HttpClientModule } from '@angular/common/http';

import { HttpClient } from '@angular/common/http';

return this.http.get(url);
```

Interceptors

```
// api.interceptor.ts

import { HttpEvent, HttpHandler, HttpInterceptor, HttpRequest }
from '@angular/common/http';
import { Observable } from 'rxjs/Observable';


export class ApiInterceptor implements HttpInterceptor {
    intercept(req: HttpRequest<any>, next: HttpHandler):
Observable<HttpEvent<any>> {
      console.log('Intercepted: ', req);
      return next.handle(req);
    }
}


// app.module.ts

providers: [
  { provide: HTTP_INTERCEPTORS, useClass: ApiInterceptor, multi:
true }
],
```

Progress Events

```
const req = new HttpRequest('GET', this.baseURL, {
  reportProgress: true
});


this.http.request(req).subscribe((event: HttpEvent<any>) => {
  switch (event.type) {
    case HttpEventType.Sent:
      console.log('Request sent!');
      break;
    case HttpEventType.ResponseHeader:
      console.log('Response header received!');
      break;
    case HttpEventType.DownloadProgress:
      const kbLoaded = Math.round(event.loaded / 1024);
      const kbTotal = Math.round(event.total / 1024);
      console.log(`Download in progress! ${ kbLoaded }Kb loaded of ${
kbTotal }Kb total`);
      break;
    case HttpEventType.Response:
      console.log('Done!', event.body);
  }
});
```

Add Service

$ ng generate service people/people –module=app

OR

$ ng g s people/people –module=app

1. Creates people.service.ts
2. Adds entry in @NgModule providers for PeopleService

Observables

Observables	Promise
Observables handle multiple values over time	Promises are only called once and will return a single value
Observables are cancellable	Promises are not cancellable

Subscribe

```
this.peopleService.getCharacters()
    .subscribe(
      (characters: Person[]) => this.characters = characters,
      (err: any) => console.log(err),
      () => console.log('Characters Retrieved: ' +
this.characters.length)
);
```

Modules

Modules

- AppModule is the “Root Module” that defines how the rest of the application will work
- Defines NgModule that has 4 sections:
  - Declarations – Components are in local scope (private visibility)
  - Imports – Modules needed by the application
  - Exports – Modules to be used by other parts of the application
  - Providers – Services that are in global scope (public visibility)
  - Bootstrap – The main (first) component to be loaded (bootstrapped)

Performance

Performace

- Compile
  - $ ng build
  - $ ng serve –prod
- Lazy Loading Modules
  - ```
    RouterModule.forRoot([
    { path: 'people', loadChildren:
    'app/people/people.module#PeopleModule' }
    ```
- Package Size
  - $ ng build –prod –sourcemaps=true
  - source-map-explorer [filename].js
    ```

Thanks

By Jeff Maxwell

TCN Solutions Inc.

jeff@tcnsolutions.com

www.jeffmaxwell.com

AngularJS (1.X) vs. Angular 2+

- The conversion from 1.X to 2+ is almost a complete rewrite
- There are some similarities but most of the code will need to be changed

If

AngularJS (1.X)

```
<table ng-if=”people.length">
```

Angular 2+

```
<table *ngIf=”people.length">
```

Loops

AngularJS (1.X)

```
<tr ng-repeat=”person in people">
  <td>{{person.name}}</td>
</tr>
```

Angular 2+

```
<tr *ngFor="let person of people">
  <td>{{person.name}}</td>
</tr>
```

Click

AngularJS (1.X)

```
<button ng-click="toggleImage()">
<button ng-click="toggleImage($event)">
```

Angular 2+

```
<button (click)="toggleImage()">
<button (click)="toggleImage($event)">
```

Model Binding

AngularJS (1.X)

```
<input ng-model=”character.name"/>
```

Angular 2+

```
<input [(ngModel)]=”character.name" />
```